

## GROUP 6

Before explaining the algorithm it's necessary to explain the two proc that are called inside the dualVth proc : one is cells\_swapping\_no\_resizing and the other one is explore\_merit.

The first one takes as input the full name of a cell and the type of the cell that I want (HVT or LVT) and ,if needed, swap the original cell with an alternative cell of the type specified. The proc doesn't change neither area nor size of the specified cell.

The second proc doesn't have any input, rather return a list of elements characterized in this way

$$\{UXYZ \quad A\}$$

where UXYZ is the full name of a cell and A is a merit parameter. To compute the merit parameter the explore merit proc get every cell in the design, takes their name, slack and leakage power; after that the proc calls the other proc (cells\_swapping\_no\_resizing) and swap every cell to HVT (it is supposed that the design given is composed only of LVT cells). After the swap the new leakage power is taken and the merit parameter is computed for every cell as follows

$$\text{merit} = 1 / \text{slack} * (\text{leakage\_as\_LVT} - \text{leakage\_as\_HVT})$$

then a list is made that link every cell full\_name to their merit parameter, the list is then ordered in decreasing order with respect to the merit parameter (the bigger the merit parameter means that either the cell has a small slack or has a small gain in terms of leakage power saved, either way it's a good candidate to be swapped back to LVT since either will impact negatively on the slack or has no real gain in terms of leakage saved).

In the main body of the proc dualVth we can see that it's divided into three parts depending on the input savings:

- if savings was 0 the explore\_merit proc will not be called and the design will have all LVT cells
- if savings was 1 the explore\_merit proc will not be called, instead every cell in the design will be swapped to HVT regardless of their merit parameter (get\_cells will give the cell collection of our design, starting from there it's easy given the cells\_swapping\_no\_resizing proc to swap every cell to HVT
- if savings was  $0 < \text{savings} < 1$  first the explore\_merit proc it's called. Starting from the ordered list returned from that proc the algorithm will swap back to LVT the cells (which are HVT after the explore\_merit proc call) one by one controlling at each step if

$$\text{old\_leakage} - \text{current\_leakage} / \text{old\_leakage} > \text{savings}$$

When that equations it's not true it means that I've swapped one too many cells back to LVT, so I just need to swap back again to HVT that last cell to obtain the final design. The final design will have a mix of LVT and HVT cells and the leakage power will satisfy the contest constraint in terms of leakage power saved.